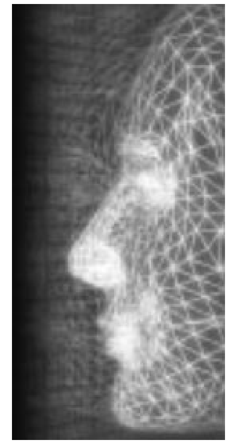
COMPUTER ANIMATION AND VIRTUAL WORLDS
Comp. Anim. Virtual Worlds 2007; **18**: 483–492
Published online 2 July 2007 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/cav.192

Learning-based 3D face detection using geometric context



By Yanwen Guo, Fuyan Zhang, Chunxiao Liu, Hanqiu Sun and Qunsheng Peng*

In computer graphics community, face model is one of the most useful entities. The automatic detection of 3D face model has special significance to computer graphics, vision, and human–computer interaction. However, few methods have been dedicated to this task. This paper proposes a machine learning approach for fully automatic 3D face detection. To exploit the facial features, we introduce geometric context, a novel shape descriptor which can compactly encode the distribution of local geometry and can be evaluated efficiently by using a new volume encoding form, named integral volume. Geometric contexts over 3D face offer the rich and discriminative representation of facial shapes and hence are quite suitable to classification. We adopt an AdaBoost learning algorithm to select the most effective geometric context-based classifiers and to combine them into a strong classifier. Given an arbitrary 3D model, our method first identifies the symmetric parts as candidates with a new reflective symmetry detection algorithm. Then uses the learned classifier to judge whether the face part exists. Experiments are performed on a large set of 3D face and non-face models and the results demonstrate high performance of our method. Copyright © 2007 John Wiley & Sons, Ltd.

Received: 15 May 2007; Accepted: 15 May 2007

KEY WORDS: 3D face model; face detection; geometric context; AdaBoost learning

Introduction

For the wide applications in biometric identification, face tracking, and human computer interaction, 2D face detection and recognition from images were intensively explored in the past decades. Many methods have been brought forward so far.[1–5] Since 2D image is prone to variations of pose, expression, and illumination, the robust and efficient techniques are still challenging. With the fast development of 3D scanning techniques, 3D model retrieval is becoming convenient. In contrast with 2D image, 3D model normally contains more inherent information for special modalities. People thus attempt to seek the solution using 3D information.[6,7] To the best of our knowledge, very few methods addressed the automatic detection of 3D face model.

3D face detection is the process of judging whether the given 3D model is or just contains the face part, and if the face part exists, locating its position on the model surface. This technology is very meaningful. For instance, when producing new characters for animation, it is often necessary to search for the available 3D faces and human models in databases or on web as reference to avoid re-scanning and re-modeling. Furthermore, the automatic detection of 3D face model will also facilitate 3D face recognition,[6] biometric identification,[7] automatic texture mapping,[8] and so on.

3D face detection involves similar issue as model retrieval, which generally refers to searching models similar to the input one from database. Current methods of model retrieval concentrate on matching global property by comparing the shapes or specifical feature descriptions of models.[9] Whereas 3D face detection is to find the local part of the given model that resembles or is exactly the face part. It is unfeasible to tackle face model detection from the point of view of model retrieval.

The most distinct property of 3D face is the geometric features of primary facial organs. The method proposed in Reference [10] is based on curvature analysis of salient facial features, however, the efficiency is relatively

*Correspondence to: Y. Guo, National Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, People's Republic of China. E-mail: ywguo@cad.zju.edu.cn

low for its vertex-wise computation. The knowledge on 2D face detection may illuminate 3D face model detection. Nevertheless, compared with 2D image, the parametric domain is in general unavailable for an arbitrary 3D model. Furthermore, different geometric models usually have different representations and take on different geometric details and complexity. How to achieve high detection efficiency independent of those uncertain ingredients is also challenging.

In this paper, we address the above issues, and propose a machine learning approach for automatic 3D face detection. The approach is capable of processing geometric models quickly and meanwhile achieves high detection rate. The main contributions of our paper are twofolds:

- *Geometric context, a novel form of shape descriptor*: We define geometric context to compactly encode geometric feature, which describes the local shape distribution in a surrounding box for each reference vertex. Under volumetric representation, a new volume encoding form called *integral volume* is used to accelerate the computation of geometric contexts. As a rich and highly discriminative descriptor, geometric context is quite suitable to 3D matching and classification problems.
- *The learning-based approach for 3D face model detection*: Without reduction, geometric contexts on model surface construct a large feature space. This leads to a machine learning approach for 3D face detection. With a set of example models which include the 3D face set and non-face set, we apply AdaBoost learning algorithm to select the most effective features and to integrate them into a strong classifier for 3D face and non-face classification. The weak classifier is defined based on the statistical multi-dimensional Gaussian distribution of geometric contexts. During detection process, a new reflective symmetry detection algorithm is first applied to extract the symmetric parts as the candidates for 3D face region. Then the learned strong classifier is imposed to these parts to determine whether 3D face exists. Experimental results show that our approach reduces the speed measurement of 3D face detection from minutes to seconds.

The rest of this paper is structured as follows. Section 'Related Work' reviews the related work. Section 'Geometric Context' elaborates the definition, property as well as the calculation of geometric context. Section 'Learning-Based 3D Face Model Detection' provides the weak classifier based on geometric context and the details of our AdaBoost learning procedure. Section 'Experiments and Analysis' presents the experimental results and quantitative analysis. Conclusion and future work are given in the last section.

Related Work

We first briefly review the relevant research on symmetry detection of 3D models, as it is a basic technique used in our paper. Then, face detection of 2D image is discussed. Finally, few relevant work on 3D face detection will be addressed.

Symmetric Detection of Geometric Model

Symmetry is essential and ubiquitous for the objects in the world and many shapes exhibit important symmetries. Some methods concentrate on finding the perfect symmetries under reflection, rotation, and translation, etc. For instance, Sun and Sherrah[11] examined the correlations in the extended Gaussian image to identify the reflective and rotational symmetries. Podolak *et al.*[12] defined the planar reflective symmetry transform, through which to detect the symmetries relative to all possible planes.

Most of the above methods focus on measuring global symmetries. But for 3D face, it may only appear as a local symmetric part of the entire model. As a consequence, we need an approach that can detect local symmetries of the object. More recently, Mitra *et al.*[13] presented an algorithm that can discover the partial and approximate symmetry of objects. The algorithm is based on matching local shape signatures of points, and extracts the global and local symmetric parts by examining the clusters in a transformation space. Theoretical analysis has proved the success rate of their algorithm. Our symmetry detection algorithm improves this method and closely accommodates it to our reflective symmetry detection.

Face Detection on 2D Image

2D face detection has been studied for many years and many achievements have been achieved so far.[1,3] See Reference [4] for a detailed survey on face detection.

Among 2D face detection methods, those based on learning have demonstrated excellent results. In 2001, Viola *et al.*[2] presented an efficient and robust method based on AdaBoost, with an image representation called integral image allowing fast Haar-like features'

DOI: 10.1002/cav

evaluation. Adaboost is a feature selection method to select discriminative features and to combine them (as weak classifiers) into a strong classifier with enhanced discriminative power.[14] Similarly in this paper, we adopt AdaBoost to construct the classifier for 3D face detection. Nevertheless, how to define efficient and discriminative geometric features are the key issues to be resolved.

3D Face Detection

Few methods were dedicated to the automatic detection of 3D face model. The method proposed by Funck *et al.*[10] is built upon the local geometry analysis around every vertex of the model. It computes the average geometric curves distributed on the face model and compares them with the studied curves to verify the face model. This algorithm's efficiency is rather low for the vertex-wise computing. As mentioned in their paper, on ordinary PC hardware, the computing time ranges from several tens of minutes to an hour according to the complexity of input models. Colombo *et al.*[15] detect the face model using curvature analysis and PCA-based classifier. The algorithm is mainly designed for onefold depth image and is unsuitable to the arbitrary input 3D model.

Geometric Context

An effective 3D shape descriptor plays an important role in 3D face model detection, it can reduce the ambiguity in detecting process. For generating a compact, yet discriminative shape descriptor, we must make the most of the predominant geometric facial features. One plausible way is to take into account the curvature features of primary facial organs. But curvature analysis normally needs vertex-wise computation, which will cut down the detection efficiency and is prone to noise in general. So for the robust and efficient detection, regional features rather than vertex features are preferred. As a key contribution, we define here 'geometric context' as a 3D shape descriptor, which encodes the local shape of geometric model by recording the shape distribution of the reference vertex in its local surrounding box. The computation of geometric context can be accelerated using a new volume encoding form, named *integral volume*.

Definition of Geometric Context

For an arbitrary vertex v_i on the input 3D model, consider the set of vectors originating from v_i to its sampled neighborhood vertices. These vectors can express the configuration of local shape at the reference vertex v_i. Obviously, the sampled neighborhood vertices get denser; the shape representation will be more exact, hence this set of vectors is a rich shape description. But the full set of vectors as a shape descriptor is much too detailed and it depends on the 3D model's representation form as well as sampling density of neighborhood vertices. So instead of the continuous and unitary representation, we identify the discrete distribution at relative positions as a more robust and compact, yet highly discriminative descriptor.

Given the 3D model, we first voxelize its surface. For every generated voxel v_i, calculate its central surrounding box with edge length R. Since this shape descriptor will only be applied to symmetric part of the input model, a fixed symmetric plane and a base plane exist (see the details in Subsection 'Reflective Symmetry Detection'). v_i's surrounding box can thus be fixed by making its two vertical side faces respectively parallel to the symmetric plane and the base plane of model surface.

We then divide v_i's surrounding box into N uniform sub-cubes, and define the number of sampling voxel v lying in v_i's nth sub-cube as 3D model's shape in v_i's nth sub-cube,

$$s_i(n) = NUM\{v, v \in \text{sub-cube}_i(n)\} \tag{1}$$

Here, $n = 1, \ldots, N$ denotes the index of nth sub-cube and v represents the voxel of 3D model surface.

We further normalize $s_i(n)$ with respect to sub-cube$_i(n)$'s full voxel volume,

$$\widetilde{s}_i(n) = \frac{s_i(n)}{VOL(\text{sub-cube}_i(n))} \tag{2}$$

The shape descriptor, *geometric context*, at voxel v_i can be defined as the following in-order array,

$$S_i = \{\widetilde{s}_i(n) | n = 1, \ldots, N\} \tag{3}$$

Figure 1 demonstrates a 2D illustration of geometric context.

In practice, the edge length of each sub-cube is valued with $R/3$, and empirically uniform subdivision of the surrounding box into $N = 27$ sub-cubes is enough to

DOI: 10.1002/cav

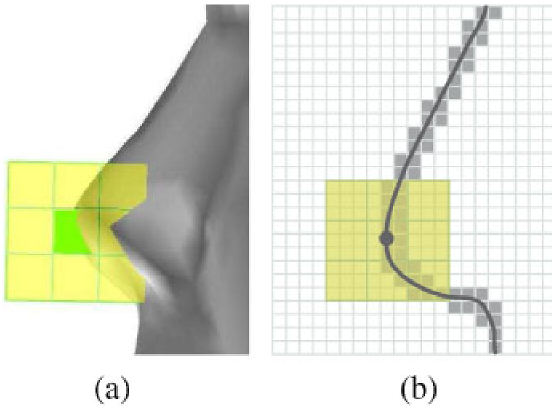


Figure 1. Geometric context. (a) The zoomed-in side view. (b) A 2D illustration. Each $\tilde{s}_i(n)$ of S_i is valued with the ratio of sub-cube$_i(n)$'s gray voxel number to sub-cube$_i(n)$'s volume of voxel.

encode geometric features. Figure 2 shows an example of such division on the nose of 3D face.

Property of Geometric Context

Similar to other geometric descriptors successfully used in 2D shape matching[16] and geometric data registration,[17,18] geometric context is a quantity computed for each surfacial voxel on the model, based on the local shape around the reference vertex. As the local surrounding box is divided and the shape ingredient is computed in each subdivision, the geometric context is not only rich and compact, but also highly discriminative to characterize facial features.

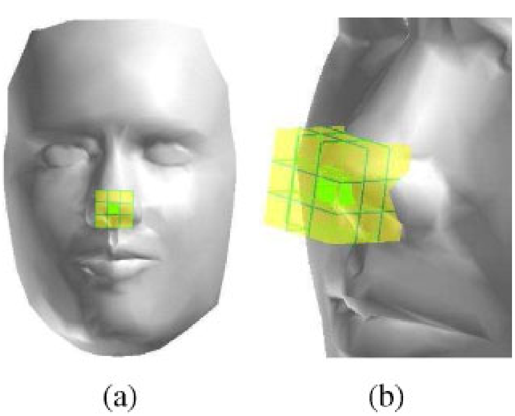


Figure 2. Division of surrounding box. (a) The frontal view of surrounding box. (b) A zoomed-in slant view of the division. v_i is enveloped in the central sub-cube.

An alternative surrounding entity of the reference point is the ball, which can be divided with respect to its radius, spherical elevation as well as rotation angles in the framework of spherical coordinate. Nevertheless, in such way, the computation of geometric context is time consuming. As can be seen in the next subsection, the geometric context described in Subsection 'Definition of Geometric Context' can be computed quickly by using *integral volume*, a new volume-encoding form.

Calculation of Geometric Context

Our 3D face detection approach needs to compute geometric context for each surfacial voxel of the symmetric model or its symmetric part. Although current voxelization algorithm[19] has achieved nearly real-time performance for complex models, computation of geometric context still undergoes much redundant processing. We introduce here *integral volume*, a new volume-encoding form to improve the efficiency, which is computed in the voxelization process.

The principle of integral volume resembles integral image that has been used in 2D face detection.[2] We first give its explanation on 2D space, then extend it to 3D. In Figure 3(a), the 2D shape contour is surrounded by one rectangle, which is divided into many 2D voxels. Every voxel is set a value V, and if the voxel is on the contour, V is set 1, otherwise 0. In Figure 3(b), every voxel is set a integral volume value IV that equals the number of all voxels lying on the contour in its upper-left area, that is,

$$IV(x_0, y_0) = \sum_{x \le x_0, y \le y_0} V(x, y) \tag{4}$$

where $V(x, y)$ is the value of voxel with discrete coordinate (x, y) in Figure 3(a).

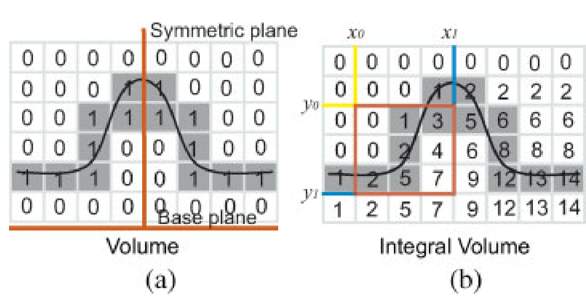


Figure 3. 2D case of integral volume. (a) The shape contour voxel is set 1, while the rest 0. (b) For every position, the integral volume stores the sum of its upper-left voxels' value.

Comp. Anim. Virtual Worlds 2007; **18**: 483–492
DOI: 10.1002/cav

Using every voxel's IV value, we can easily calculate the contour voxels' number included in any sub-rectangle. For example, the voxels' number in rectangle (Figure 3(b)) with upper-left point (x_0, y_0) and bottom-right point (x_1, y_1) is

$$IV(x_0, y_0, x_1, y_1) = IV(x_1, y_1) + IV(x_0, y_0) - IV(x_0, y_1) - IV(x_1, y_0) \quad (5)$$

Similarly, for every voxel on 3D model surface, its value V and integral volume value IV can be defined as follows,

$$IV(x_0, y_0, z_0) = \sum_{x \le x_0, y \le y_0, z \le z_0} V(x, y, z) \quad (6)$$

Now back to Subsection 'Definition of Geometric Context,' $s_i(n)$, that is the surfacial voxels' number in nth sub-cube, can be easily obtained. Assuming that the nth sub-cube has two diagonal vertices (x_0, y_0, z_0) and (x_1, y_1, z_1), then,

$$\begin{aligned} s_i(n) &= IV(x_0, y_0, z_0, x_1, y_1, z_1) \\ &= IV(x_1, y_1, z_1) + IV(x_1, y_0, z_0) \\ &\quad - IV(x_1, y_1, z_0) - IV(x_1, y_0, z_1) \\ &\quad - IV(x_0, y_1, z_1) + IV(x_0, y_1, z_0) \\ &\quad - IV(x_0, y_0, z_0) + IV(x_0, y_0, z_1) \end{aligned} \quad (7)$$

Obviously, with the above formula, the voxel v_i's geometry context can be obtained easily.

In summary, the integral volume in fact defines a searching table in the surrounding box. By means of it, we can easily obtain the number of surfacial voxels in each sub-cube. The computation of geometric contexts with different edge lengths R over model surface is thus very fast. In the following, we introduce the approach of 3D face detection with geometric context.

Learning-Based 3D Face Model Detection

Our basic observation is that 3D face part is reflective symmetric about a fixed plane. So the first step is to extract the reflective symmetric and nearly reflective symmetric parts in the input 3D model, through which we reject those non-face regions and discover the candidates for the face part as well. For each extracted symmetric part, the geometric contexts over its voxelized surface form a large space of classifiers. We propose to use AdaBoost learning to select the most effective classifiers to construct a strong classifier, through which the symmetric part is judged whether it is or just contains the 3D face. We first describe our algorithm for identifying and extracting the reflective symmetric parts.

Reflective Symmetry Detection

3D face region is reflective symmetric. That is, it is unchanged by reflecting about the symmetric plane P. In particular, for each vertex v_i on the face part, its reflected vertex v_j about P exists on the face. Furthermore, for such vertex pair (v_i, v_j), two symmetric conditions should be satisfied,

- some of their intrinsic geometric properties, for example, mean and Gaussian curvatures should be equal;
- Other geometric properties, for example normal vectors, principle directions should be equivalent under reflection about P.

Our algorithm for detecting reflective symmetry is built upon searching possible symmetric vertex pairs, and accumulating the evidence of the symmetric plane P.

We densely sample the 3D model, and, for any vertex pair (v_i, v_j), judge if they are likely to be symmetric by checking the above two symmetric conditions. If so, we record their potential symmetric plane P_{ij} using a quad $Q_{i,j} = (a, b, c, d)$, where $ax + by + cz + d = 0$ is the plane equation with $d = \{0, 1\}$. P_{ij} passes through the center of v_i and v_j and meanwhile it is perpendicular to $v_i v_j$. For efficiency, we select the feature vertices of 3D model as the sampled vertex pairs to detect symmetric plane. For instance, we can select those vertices whose mean curvatures exceed a given threshold. For the sake of noise, more complex descriptors, for example, integral sphere[18] instead of the curvature can be used in the symmetry detection process.

We then assemble all possible symmetric vertex pairs. Their corresponding quads $\{Q_{i,j}\}$ form a space. Obviously, those quads in this space that potentially correspond to the real symmetric plane will be close enough. We can therefore cluster these close quads to extract the symmetric plane (Figure 4).

Note that, each symmetric plane detected needs to be further verified by testing whether its supporting vertices are spatially adjacent on the model. Through such verification, we simultaneously extract the approximate symmetric area. Recall that a base plane is needed

DOI: 10.1002/cav

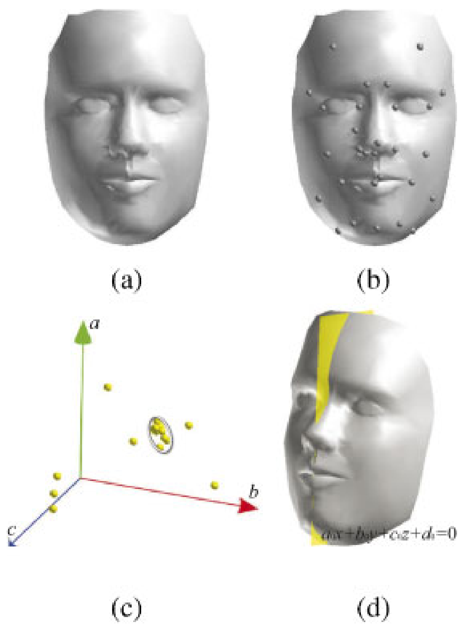


Figure 4. Symmetry detection. (a) The 3D model; (b)sampled feature points. (c) Quads space for those potentially symmetric feature points. Some quads in the circle cluster together. (d) The extracted symmetric plane with equation $a_0x + b_0y + c_0z + d_0 = 0$, $d_0 = 0, 1$.

to construct geometric context. We take the one that is perpendicular to the symmetric plane and passes through the first eigenvector produced with principle component analysis on sampled vertices as the base plane. The algorithm for reflective symmetric detection is rather fast and effective.

AdaBoost Learning-Based 3D Face Model Detection

Through the above reflective symmetry detection algorithm, asymmetric regions are eliminated and only those reflective symmetric and nearly reflective symmetric parts are viewed as the candidates for the 3D face. We then examine those parts and determine whether the face part exists using AdaBoost learning algorithm.

With voxel representation, the geometric contexts with variant edge length R over surface of the symmetric part construct a large feature space. Each geometric context can perform simple classification of 3D face and non-face, that is, it yields a weak classifier. Nevertheless, no single classifier can achieve high detection rate reliably and accurately. In addition, not all the classifiers are crucial. Most of them exert trivial effect in general. As a result, based on a great deal of training examples which include 3D face models and non-face models, AdaBoost learns the weak classifiers and combines the most effective ones to form a strong classifier.

AdaBoost is an iterative process. During every iteration, the classifiers are trained and the one with the lowest classification error is picked out. Thereafter, those difficult examples of classification will be emphasized in the following iterations by assigning them higher training weights. The selected classifiers are finally integrated to form a strong classifier with the learned weighted coefficients.

Weak Learning. A fundamental step of AdaBoost iteration is the weak learning algorithm, which is devised to select the weak classifier that can best separate the 3D face and non-face models. Since the geometric context is N-dimensional, the weak classifier is built upon the distribution of the 3D face models' corresponding geometric contexts.

As a preprocessing, we first align the examples of 3D face models together with the non-face models into consistent reference frame. Then we voxelize their surfaces into uniform voxel representation and compute the geometric contexts separately.

For the corresponding surfacial voxels at the same position on 3D face examples, we assume that every element $\tilde{s}_i(n)$ of their geometric contexts with consistent edge length follows Gaussian distribution. Hence the N-dimensional geometric contexts S satisfy N-dimensional Gaussian distribution,

$$G_i(S) = \alpha_{S_i} e^{-\frac{1}{2}(S-m_{Si})^T C_{S_i}^{-1}(S-m_{Si})} \quad (8)$$

where i represents the index of the current voxel position. m_{Si} is the mean vector, C_{Si} for the covariance matrix, and α_{Si} for a constant which gives unit normalization.

With the above distribution, the weak classifier h_i is defined as

$$h_i(S) = \text{sign}(G_i(S) - \theta_i) \quad (9)$$

During the weak learning, θ_i is chosen as a threshold such that the least examples are misclassified.

AdaBoost Learning Algorithm. Given the set M of training examples which consist of K models and their labels as following,

$$M = (x_j, y_j)_{j=1,\dots,K} \quad (10)$$

Comp. Anim. Virtual Worlds 2007; **18**: 483–492
DOI: 10.1002/cav

Give the training examples M.
For $t = 1, \ldots, T$:

- Normalize the weights $w_{t,j\, j=1,\ldots,K}$,
$$\sum_{j=1}^{K} w_{t,j} = 1.$$
- Train the classifiers, each of which h_i is restricted to using a single feature.
- Evaluate the error with respect to $w_{t,j}$,
$$\epsilon_i = \sum_{j=1}^{K} w_{t,j} |h_i(x_j) - y_j|$$
- Choose h_t with the lowest error ϵ_t and set:
$$\alpha_t = \frac{1}{2}\log\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$$
- Update the weights:
$$\omega_{t+1,j} = \omega_{t,j}\exp(-\alpha_t y_j h_t(x_j))$$

The final strong classifier is,
$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$

Table 1. Algorithm: AdaBoost learning

where x_j is the j input model and y_j is the corresponding label. If x_j is the 3D face model, y_j is set 1; otherwise -1. Every training model is also associated with one initial weight $w_{1,j\, j=1,\ldots,K}$. If $y_j = 1$, $w_{1,j} = 1/2M$, else $1/2L$. Here M and L are the number of 3D face models and non-face models.

AdaBoost learning is an iterative process, we suppose there are totally T iterations. The strong classifier can be obtained after T iterations. Table 1 summarizes the algorithm.

Face Detection Process. During detection, the selected weak classifiers are orderly imposed on each extracted symmetric part. For current symmetric part, the final value is calculated by combing the value output of each classifier h_t using the learned weights α_t. If the final value $H(x)$ is bigger than a given threshold, the symmetric part is viewed as the 3D face part, otherwise non-face part.

Experiments and Analysis

In this section, we describe the learning details with a batch of models and demonstrate detection performance tested by a model database using AdaBoost learning.

Data Description

The models used for training consist of two parts: face models and non-face models. In them, the training face models come from three databases. The first one is USF HumanID 3DFS-100-3 face database.[20] Total 600 3D face modes are synthesized by the morphable model described in Reference [20] based on the principal component analysis of 100 exemplar 3D faces. The second one consists of some face models from University of Washington, which are generated using the method described in Reference [21]. This database includes 384 face meshes with different expressions, 66 models of them are selected to train classifiers. The third database includes 92 models downloaded from the public database and most of them are generated using 3D scanners or existing modeling softwares. All the 3D faces are aligned by first rotating them into consistent orientation and then scaling them so that the distances between left and right canthi are the same.

In addition, we have 562 non-face models, which are symmetric ones and nearly symmetric disc-topology model patches (Figure 5).

All the above training models are further laid out on a common plane vertical to their medial symmetric planes

Figure 5. Training data. The first four rows: some 3D face models. The last two: representative non-face models.

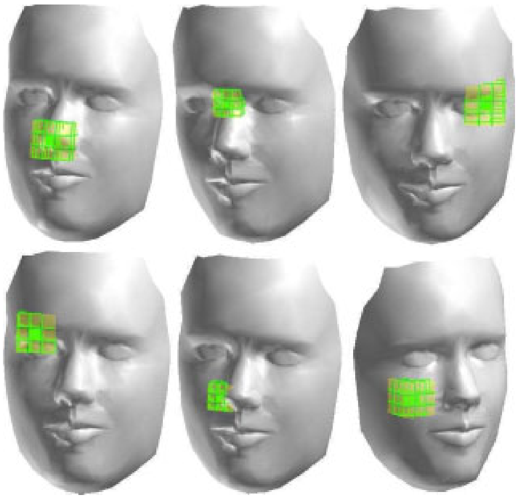


Figure 6. The first six most discriminative features selected by AdaBoost.

and fit into a common frame. Thereafter, their surfaces are voxelized in the frame with uniform voxel size and consistent orientation. Finally, geometric contexts with a set of variant edge length R over surface voxels are calculated for face detection using Adaboost learning.

Face Detection

The Adaboost learning algorithm described in Section 'Learning-Based 3D Face Model Detection' is applied to the training data. During this process, 162 most discriminative classifiers are selected. Figure 6 shows Six features of them. As can be seen these features encode the primary facial features, for example, nose, canthus, etc.

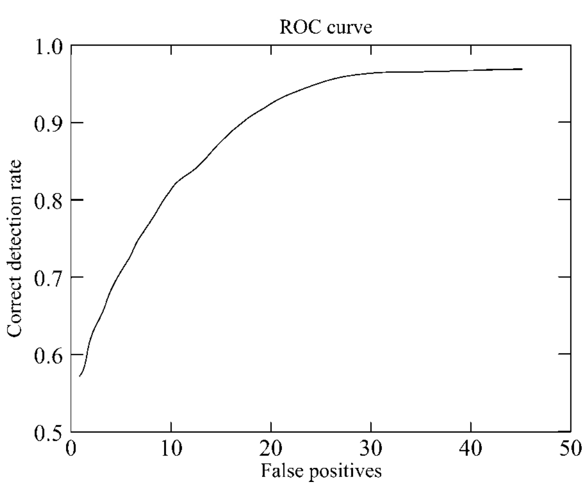


Figure 7. Roc curve for our detector on test set.

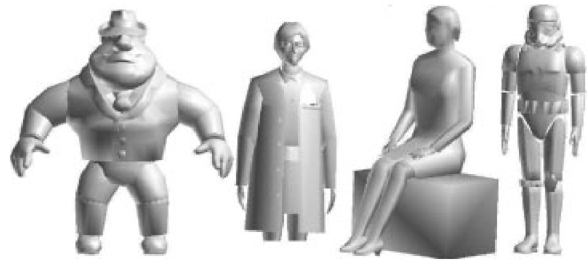


Figure 8. Some models that cannot be detected.

This verifies our subjective perception, that is, geometric features of primary facial organs are the most distinct characteristics that 3D faces differ from other models.

We further test the detection accuracy of the learned classifiers on a probe set, which includes 532 3D faces and 1000 non-faces. Among them, half of the 3D faces still are synthesized using USF HumanID 3D face database with different parameters; some are scanned using a 3Space Fast Scan hand held scanner, while all the rest models come from the public NTU model database.[22]

The detection time for a geometric model normally takes less than 10 seconds on a standard PC with an Intel PIV 3.0 GHz CPU and 1G memory. Figure 7. shows the receiver operator characteristic (ROC) curve of our detector. In addition, nearly 20 3D face or human models in NTU database can not be detected due to influence of cap, glasses, or lack of features on 3D faces (Figure 8)

Conclusions and Future Work

This paper presents an efficient 3D face detection approach based on AdaBoost learning. To compactly encode the facial geometric features, we introduce geometric context, a novel 3D shape descriptor, whose computation speed over model surface can be very fast with integral volume. AdaBoost is employed to select the most discriminative geometric context-based features, and to integrate them into a strong classifier for 3D face and non-face classification.

Generally, the detection rate of learning-based algorithm depends heavily on the training examples, for example, the number of examples as well as different race. We intend to collect more 3D face models to enrich the training examples and to enhance the detection performance.

Our 3D face detection approach can be behaved as a preliminary step for 3D face recognition and biometric identification, and can also be extended to the automatic

Comp. Anim. Virtual Worlds 2007; **18**: 483–492
DOI: 10.1002/cav

detection of geometric facial features for applications in facial expression cloning and editing. In addition, as a rich, compact yet discriminative shape descriptor, geometric context can be further investigated to patch registration of geometric data. In this way, the optimized pairwise matching for different sets of geometric contexts needs to be meticulously considered. We will investigate it in future.

ACKNOWLEDGMENTS

The work is supported by CUHK Direct Research Grant (No. 2050349), National 973 program (No. 2002CB312101), and NSFC grant (No. 60403038) in China. The authors thank Dr Feng Tang for his helpful discussion and the anonymous reviewers for their thoughtful comments. Models are courtesy of the University of South Florida, University of Washington Graphics and Imaging Laboratory, National Taiwan University, max-planck-institut informatik and the aim@shape database.[23] All the models listed in this paper are used for academic research only.

References

1. Rowley HA, Baluja S, Kanade T. Neural network-based human face detection. *IEEE Transactions on PAMI* 1998; **20**(1): 23–38.
2. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2001; 511–518.
3. Zhang Z, Li SZ, Zhang H. Real-time multi-view face detection. In *Conference on Automatic Face and Gesture Recognition*, 2002; 149–154.
4. Yang MH, Kriegman D, Ahuja N. Detecting faces in images: a survey. *IEEE Transactions on PAMI* 2002; **24**(1): 34–58.
5. Phillips PJ, Moon H, Rizvi S, Rauss P. The feret evaluation methodology for face-recognition algorithms. *IEEE Transactions on PAMI* 2000; **22**(10): 1090–1104.
6. Gordon GG. Face recognition based on depth and curvature feature. 1992; 108–110.
7. Chang KI, Bowyer KW, Flynn PJ. An evaluation of multi-model 2d+3d biometrics. *IEEE Transactions on PAMI* 2005; **27**(4): 619–624.
8. Guo YW, Wang J, Sun HQ, Peng QHS. A novel constrained texture mapping method based on harmonic map. *Computers & Graphics* 2005; **29**(6): 972–979.
9. Funkhouser T, Min P, Kazhdan M, *et al*. A search engine for 3d models. *ACM Transactions on Graphics* 2003; **22**(1): 83–105.
10. Von Funck W, Theisel H, Seidel HP. Shape matching based on fully automatic face detection on triangular meshes. In *Computer Graphics International*, vol. 4035. pp. 242–253, 2006.
11. Sun C, Sherrah J. 3d symmetry detection using extended Gaussian image. *IEEE Transactions on PAMI* 1997; **19**(2): 164–168.
12. Podolak J, Shilane P, *et al*. A planar-reflective symmetry transform for 3d shapes. *ACM Transactions on Graphics* 2006; **25**(3): 549–559.
13. Mitra NJ, Guibas LJ, Pauly M. Partial and approximate symmetry detection for 3d geometry. *ACM Transactions on Graphics* 2006; **25**(3): 560–568.
14. Freund Y, Schapire R. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* 1997; **55**: 119–139.
15. Colombo A, Cusano C, Schettini R. 3d face detection using curvature analysis. *Pattern Recognition* 2006; **39**(3): 444–455.
16. Belongie S, Malik J, Puzicha J. Shape matching and object recognition using shape contexts. *IEEE Transactions on PAMI* 2002; **24**(4): 509–522.
17. Huber D, Hebert M. Fully automatic registration of multiple 3d data sets. *Image and Vision Computing* 2003; **21**(7): 637–650.
18. Gelfand N, Mitra NJ, Guibas LJ, Pottmann H. Robust global registration. In *Symposium on Geometry Processing*, 2005; pp. 197–206.
19. Dong Z, Chen W, Bao HJ, Zhang HX, Peng QHS. Real-time voxelization for complex polygonal models. In *Pacific Graphics*, 2004; pp. 73–78.
20. Blanz V, Vetter T. Morphable model for the synthesis of 3d faces. In *Siggraph 99*, pp. 187–194, 1999.
21. Zhang L, Snavely N, Curless B, Seitz S. Spacetime faces: high-resolution capture for modeling and animation. *ACM Transactions on Graphics* 2004; **23**(3): 548–558.
22. http://3d.csie.ntu.edu.tw/∼dynamic/databa-se/index.html
23. http://www.aimatshape.net

Authors' biographies:

Yanwen Guo born in 1980, is now an Assistant Professor at the National Laboratory for Novel Software Technology, Nanjing University, P.R. China. He received his Ph.D. in State Key Lab of CAD&CG, Zhejiang University in 2006. His research interests include real-time computer graphics, image video processing, and computer vision.

Fuyan Zhang is now a Professor at the National Laboratory for Novel Software Technology, Nanjing University, P.R. China. His research interests include computer graphics, digital library, multimedia computing, and system.

DOI: 10.1002/cav

Chunxiao Liu born in 1979, Ph.D. candidate at the State Key Lab. of CAD&CG, Zhejiang University. His research interests include image and video completion, image and video-based rendering, and virtual reality.

Hanqiu Sun is now an Associate Professor of Department of Computer Science & Engineering, The Chinese University of Hong Kong. She received her Ph.D. in Computer Science from University of Alberta, Canada. Her current research interests include virtual & augmented reality, hypermedia, computer-assisted surgery, etc.

Qunsheng Peng received his Ph.D. from University of East Anglia in 1983 and is currently a Professor at the State Key Lab. of CAD&CG, Zhejiang University. His research interests include virtual reality, realistic image synthesis, infrared image synthesis, computer animation, and scientific data visualization.

Comp. Anim. Virtual Worlds 2007; **18**: 483–492
DOI: 10.1002/cav